STAT451 Project

Binary pixel image

Yanzi Jin
Niloufar Dousti Mousavi
Wangfei Wang

Abstract

In this project, we developed and implemented Gibbs sampling for restoring a noisy gray-scale image. We compared the restored images with different observed data that were sampled as initial starting images, i.e., the original image with Gaussian scheme, and the image equal to the true posterior mean pixel color. Additionally, we compared the Gibbs samplers with different parameter settings, specifically with different Markov neighborhood order and variance of the model. We have shown that the initial starting images sampled is not playing a significant role in image restoration, and the difference between differing choice of neighborhood is suttle; however, the variance of the model influences the sampled images greatly.

1 Introduction

Markov chain Monte Carlo (MCMC) has become a popular method in solving complicated statistical problems. MCMC is particularly useful in Bayesian analysis, where complex and high dimensional integrals are involved in obtaining posterior distributions [1]. One application of MCMC is Bayesian analysis of Markov random field (MRF), which is considered as a set of random variables that have Markov property. A MRF can be regular or irregular, and based on the individual sites and associated random variables, it can be further classified into different groups [2]. MRF is extensively used to model studies in imaging processing.

Bayesian image analysis includes the removal of noise of an image, the generation of better image from an original noisy image, the restoration of multi-dimensional images from lower dimensional images and etc. Besag *et al.* pointed out the often neglected area in Bayesian image analysis that is indeed useful and important – image restoration [3]. Our project explores the application of this particular practical importance.

Bayesian image analysis uses probability models to incorporate the specific prior knowledge into a defined image. Hammersley and Clifford Theorem [4], first proved in an unpublished work by John Hammersley and Peter Clifford in 1971, is the key to construct proper model through the conditional probability distribution in MRF-related image restoration problems. Using this theorem, we can specify the joint distribution of pixels to a normalizing constant [5].

There are a variety of schemes for lattice systems, among which binary, Gaussian, auto-binomial, auto-Poisson, and auto-exponential schemes are quite common. Julian Besag have done extensive studies in the field, and more detailed review of each of the case can be read in [2]. Our project is a Gaussian scheme, which models the joint distribution to a multivariate normal distribution.

The central problem in MRF is that there is no direct method to simulate general multivariate distributions [2], so algorithm that is based on the corresponding univariate local information is quite helpful. Our project deals with one of the easiest scenarios in MRF, where the lattice is regular and the pixel image is as small as 20×20, which in real life is unrealistic to encounter. However, even with this small number of pixels, it could already create an enormous number of terms in the conditional probability calculation. Hence, MCMC can be used in this case to overcome the difficulty of computational burden.

In particular, Gibbs sampling, one of the MCMC tools which updates the component according to the current local information ensures the convergence to the joint distribution

under general conditions [2], and therefore is widely used in solving MRF-related problems. Sampling from multivariate densities in Gibbs sampling is pretty straigtforward as long as we can derive the conditional probability densities. Generally, we first treat the other variables as fixed in the joint probability densities, then we investigate how to sample from the conditional distribution.

In our project, the prior and likelihood are given. The prior distribution of the observed data is a Gaussian distribution. We apply Gibbs sampling to generate a collection of images from the posterior distribution with known prior density and likelihood from a 20×20 pixel image. We also compare the images generated from Gibbs samplers with differing $\mathbf{x}^{(0)}$, σ, and neighborhood structure.

2 Gibbs Sampling

2.1 Algorithm

Here, we give a brief description of Gibbs sampling: for each pixel, the current value is replaced by a new value that is randomly sampled from the conditional posterior distribution given all other current pixel values $\mathbf{x}_{-i}$ and $\mathbf{y}$. Algorithm 1 shows the pesudo code of our method. The parameter of the method is the observed image $\mathbf{x}$, the maximal iteration number, the neighborhood order, and the standard deviation σ.

We use two different initial starting images $\mathbf{x}^{(0)}$. One is the original image with Gaussian noise, and the other is the true mean value of the original image, which is equal to 57.5. The maximal number of iteration is 100 in each of the case. The order of neighborhood is either first-order or second-order. The tricky part is when getting the neighborhood, we have to handle the pixels on the edge with caution.

Algorithm 1 Gibbs sampling for image restoration.

function ImageRestoration($\mathbf{x}, T, d, \sigma$)
  for $t = 1$ to T **do**
    for x_i in $\mathbf{x}$ **do**
      Get neigborhood δ_i for pixel x_i.
      Compute the number of neighboring pixels v_i.
      Compute mean of δ_i: $\bar{x}_{\delta_i} = \frac{1}{v_i} \sum_{j \in \delta_i} x_j$.
      Sample a value following the distribution and update x_i:

$$f(x_i | \mathbf{x}_{-i}, \mathbf{y}) = \mathcal{N}\left(\frac{1}{v_i + 1} y_i + \frac{v_i}{v_i + 1} \bar{x}_{\delta_i}, \frac{\sigma^2}{v_i + 1} \right).$$

    end for
  end for
end function

2.2 Derivation

Let $\mathbf{x}$ be the pixels in the image, $\mathbf{x} = \{x_i\}$, where $i = 1, \ldots, n$. In our case, since we have a 20×20 image, $n = 400$. δ_i is the neighborhood of the ith pixel x_i, v_i is the number of pixels in the neighborhood δ_i. $\bar{x}_{\delta_i}$ is the mean value of all the pixels in the neighborhood $\mathbf{x}_{\delta_i}$. $\mathbf{y} = \{y_i\}$, where $i = 1, \ldots, n$, is the observed image. According to the model, we have the prior distribution of the true image:

$$f(x_i|\mathbf{x}_{\delta_i}) = \mathcal{N}\left(\bar{x}_{\delta_i}, \frac{\sigma^2}{v_i}\right), \quad i = 1, \ldots, n. \tag{1}$$

and the likelihood of the observed data y_i:

$$f(y_i|x_i) = \mathcal{N}\left(x_i, \sigma^2\right), \quad i = 1, \ldots, n. \tag{2}$$

Generally, we know for arbitrary unknown variable $\mathbf{x}$ and $\mathbf{y}$, assumed as a D-dimenstional vector, follow distributions:

$$f(\mathbf{x}) = \mathcal{N}\left(\mu, \Lambda^{-1}\right), \quad f(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{x}, L^{-1}), \tag{3}$$

where μ and Λ^{-1} is the mean and covariance matrix of $\mathbf{x}$, and L^{-1} is the covariance matrix of $\mathbf{y}$ conditioned on $\mathbf{x}$. Then using conditional Gaussian and Baysian rule, we have

$$\begin{aligned} f(\mathbf{x}|\mathbf{y}) &= \frac{f(\mathbf{y}|\mathbf{x}) \cdot f(\mathbf{x})}{f(\mathbf{y})} \\ &= \mathcal{N}\left((\Lambda + L)^{-1}(\Lambda\mu + L\mathbf{y}), (\Lambda + L)^{-1}\right). \end{aligned} \tag{4}$$

In our case, $\mu = \bar{x}_{\delta_i}, \Lambda = \frac{v_i}{\sigma^2}, L = \frac{1}{\sigma^2}$. Note they are real values since x_i is a univariate variable. With the Markov property, $f(x_i|\mathbf{x}_{-i}) = f(x_i|\mathbf{x}_{\delta_i})$. Therefore, we have

$$\begin{aligned} f(x_i|\mathbf{x}_{-i}, y_i) &\propto f(x_i|\mathbf{x}_{-i}) f(y_i|x_i) \\ &= f(x_i|\mathbf{x}_{\delta_i}) f(y_i|x_i) \\ &= \mathcal{N}\left[\left(\frac{v_i}{\sigma^2} + \frac{1}{\sigma^2}\right)^{-1} \cdot \left(\frac{v_i}{\sigma^2} \cdot \bar{x}_{\delta_i} + \frac{1}{\sigma^2} \cdot y_i\right), \left(\frac{v_i}{\sigma^2} + \frac{1}{\sigma^2}\right)^{-1}\right] \\ &= \mathcal{N}\left(\frac{1}{v_i + 1} y_i + \frac{v_i}{v_i + 1} \bar{x}_{\delta_i}, \frac{\sigma^2}{v_i + 1}\right). \end{aligned} \tag{5}$$

The aforementioned part is a brief derivation of the univariate conditional posterior distribution used for Gibbs sampling. A more detailed explanation is shown in **Appendix**.

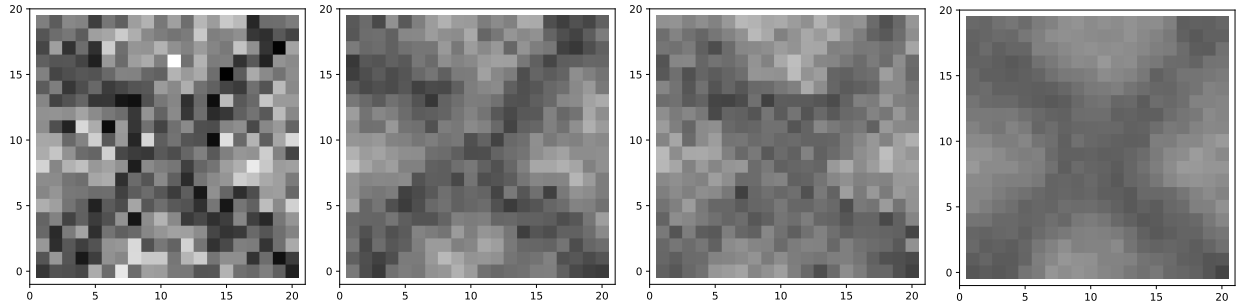
3 Results

3.1 $\sigma = 5$ and a Second-order Neighborhood

The first image sampled from the conditional posterior distribution $\mathbf{x}^{(1)}$ is shown in Figure 1b and the last image sampled from the conditional posterior distribution $\mathbf{x}^{(100)}$ is shown in

Figure 1c. The data image (in this case is also $\mathbf{x}^{(0)}$) and mean image are shown in Figure 1a and Figure 1d.

We can see that all the images have the X shape as is in the data image. However, $\mathbf{x}^{(1)}$, $\mathbf{x}^{(100)}$ did not recover all the information from the data image. For example, the pixel (19, 17) in the data image has a big value, making the pixel dark, but neither $\mathbf{x}^{(1)}$ nor $\mathbf{x}^{(100)}$ has a big value at pixel (19, 17). All $\mathbf{x}^{(1)}$, $\mathbf{x}^{(100)}$ and mean image have less contrast than the data image. The mean image is smoother than $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(100)}$, since it averages all the 100 iterated images.



(a) $\mathbf{x}^{(0)}$ (b) $\mathbf{x}^{(1)}$ (c) $\mathbf{x}^{(100)}$ (d) Mean Image

Figure 1: Images obtained from a Gibbs sampler ($\sigma = 5$ and a second-order neighborhood), with $\mathbf{x}^{(0)}$ as the observed data image, $\mathbf{x}^{(1)}$, and $\mathbf{x}^{(100)}$ as result from the first and last iteration.

3.2 2 × 3 Factorial Design

A 2 × 3 factorial design was conducted to compare the performance of various Gibbs samplers with different neighborhood order and σ (Table 1). The illustration of neighborhood structures is shown in Figure 2.

Table 1: 2 × 3 Factorial Design

	Factor B σ		
Factor A Neighborhood structure	j = 1 $\sigma = 2$	j = 2 $\sigma = 5$	j = 3 $\sigma = 15$
i = 1 first-order	Figure 3	Figure 4	Figure 5
i = 2 second-order	Figure 6	Figure 1	Figure 7

3.2.1 Factor: σ

First, we fix Factor A (neighborhood structure), and study how σ effect the Gibbs sampler. When we use the same neighborhood structure, we are fixing the conditional posterior distribution's mean, and the only thing changes is the variance of the conditional posterior normal distribution. With larger σ, the Gibbs sampler samples larger width of a particular pixel's neighbors, meaning values of distant pixels can influence the sampled pixel.

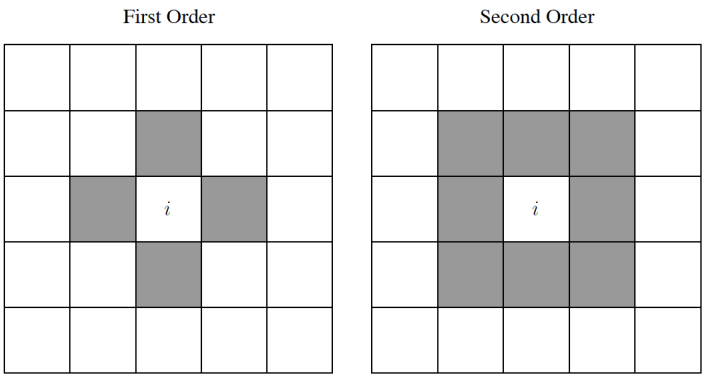


Figure 2: Shaded pixels show a first-order (left) and a second-order neighborhood (right) in a rectangular lattice. (Figure adapted from text book **FIGURE 8.6** [5]).

When first-order neighborhood is incorporated (Figure 3, 4 and 5), Figure 5 with $\sigma = 15$ has a very steep gradient between pixels (high contrast), but has the least accurate shape of X because the distance pixels influence the sampled pixel. In contrast, images rendered from $\sigma = 2$ have the least contrast among the three figures because the variance in conditional posterior distribution is small and Gibbs sampler depends more on the information from close neighborhoods. Figure 4 seems to be the best among these three figures.

In the case of second-order neighborhood, Figure 7 has the highest contrast, while the images appear to be scrambled and X shape is totally lost. Figure 6 seems to be oversmoothed. Figure 1 seems to be the best among the three sets of images sampled from second-order neighborhood.

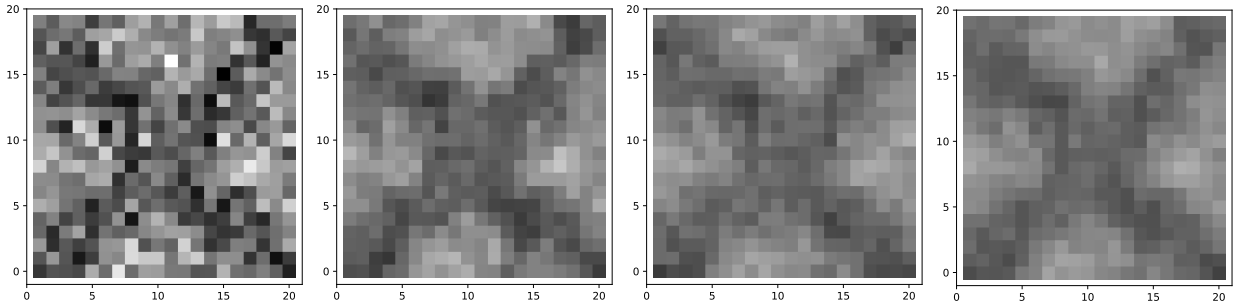
3.2.2 Factor: Neighborhood Structure

Now, we investigate how the selection of neighborhood structure influence the images rendered by Gibbs sampler. Second-order neighborhood has 8 neighbors and first-order neighborhood only has 4 neighbors (Figure 2). The neighborhood structure influences both mean and variance of the conditional posterior distribution. The second-order neighborhood has smaller variance, while the value of mean depends on the data.

The images below show that the images sampled from second-order neighborhood are smoother than those from first-order neighborhood. For example, images sampled from first-neighborhood (Figure 3) seems to have higher contrast and are slightly better than the second-neighborhood (Figure 6). However, the differences between the images sampled using different neighborhood structures are not as large as those between images sampled using different σ's.

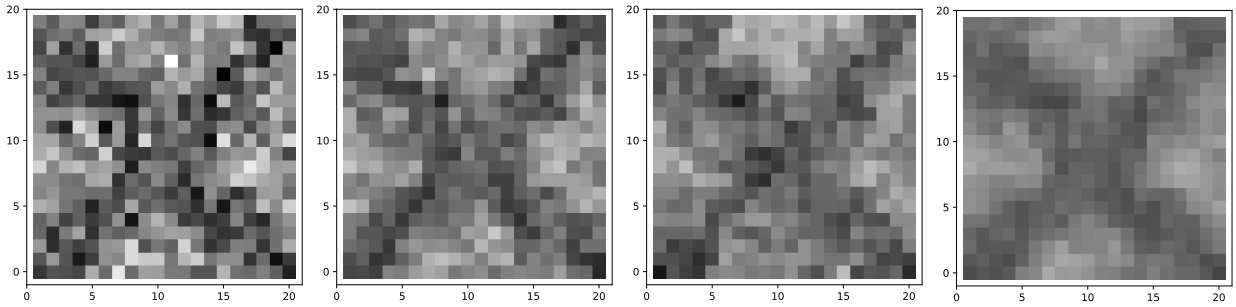
3.3 Different Starting $\mathbf{x^{(0)}}$

$\mathbf{x}^{(0)} = 57.5$, $\sigma = 5$ and first-neighborhood were used for the Gibbs sampler. The result is shown in Figure 8. This Gibbs sampler uses the initial starting image $\mathbf{x}^{(0)}$ equal to the true posterior mean pixel color. The results are pretty nice, neither too smooth, nor having too much contrast. The images are actually pretty close to those obtained by Gibbs sampler



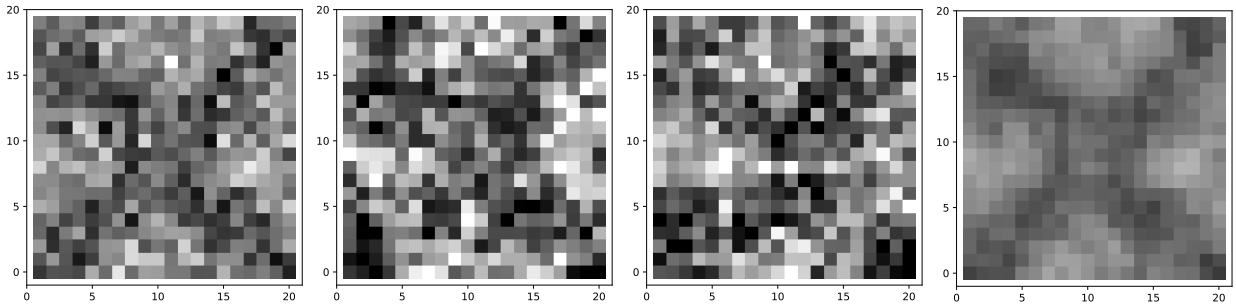
(a) $\mathbf{x}^{(0)}$ (b) $\mathbf{x}^{(1)}$ (c) $\mathbf{x}^{(100)}$ (d) Mean Image

Figure 3: Images obtained from a Gibbs sampler ($\sigma = 2$ and a first-order neighborhood), with $\mathbf{x}^{(0)}$ as the observed data image, $\mathbf{x}^{(1)}$, and $\mathbf{x}^{(100)}$ as result from the first and last iteration.



(a) $\mathbf{x}^{(0)}$ (b) $\mathbf{x}^{(1)}$ (c) $\mathbf{x}^{(100)}$ (d) Mean Image

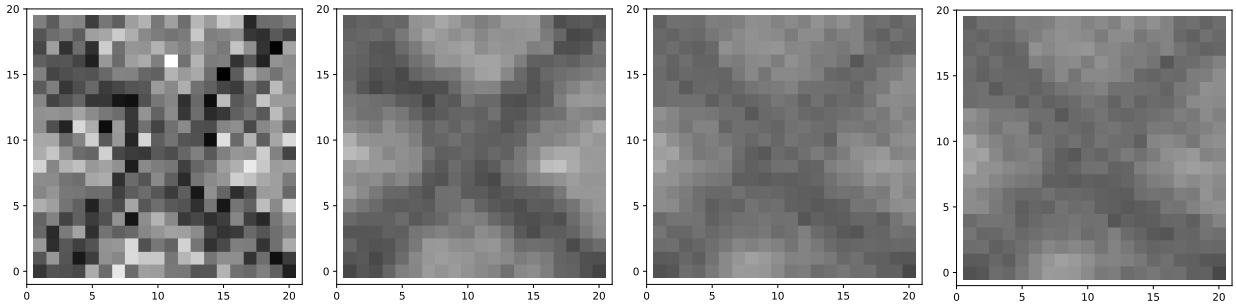
Figure 4: Images obtained from a Gibbs sampler ($\sigma = 5$ and a first-order neighborhood), with $\mathbf{x}^{(0)}$ as the observed data image, $\mathbf{x}^{(1)}$, and $\mathbf{x}^{(100)}$ as result from the first and last iteration.



(a) $\mathbf{x}^{(0)}$ (b) $\mathbf{x}^{(1)}$ (c) $\mathbf{x}^{(100)}$ (d) Mean Image

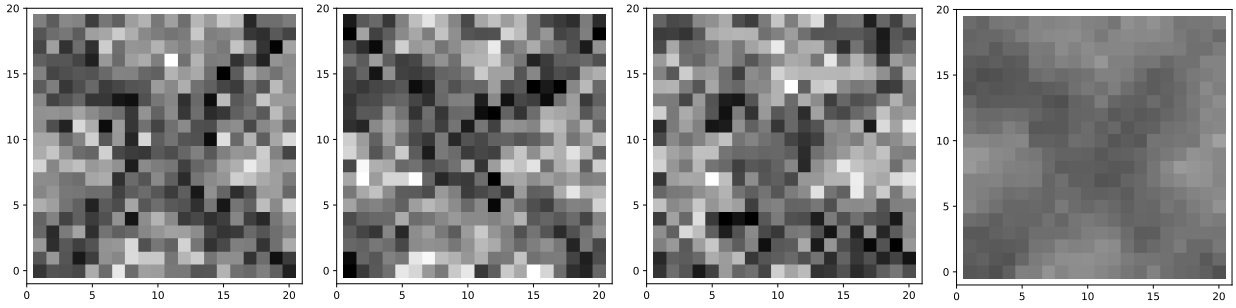
Figure 5: Images obtained from a Gibbs sampler ($\sigma = 15$ and a first-order neighborhood), with $\mathbf{x}^{(0)}$ as the observed data image, $\mathbf{x}^{(1)}$, and $\mathbf{x}^{(100)}$ as result from the first and last iteration.

with $\mathbf{x}^{(0)}$ = the observed data image, $\sigma = 5$ and a first-order neighborhood (Figure 4). This indicates that the Markov chain moves fast from the initial state. Therefore, initial starting $\mathbf{x}^{(0)}$ is not very important in this case.



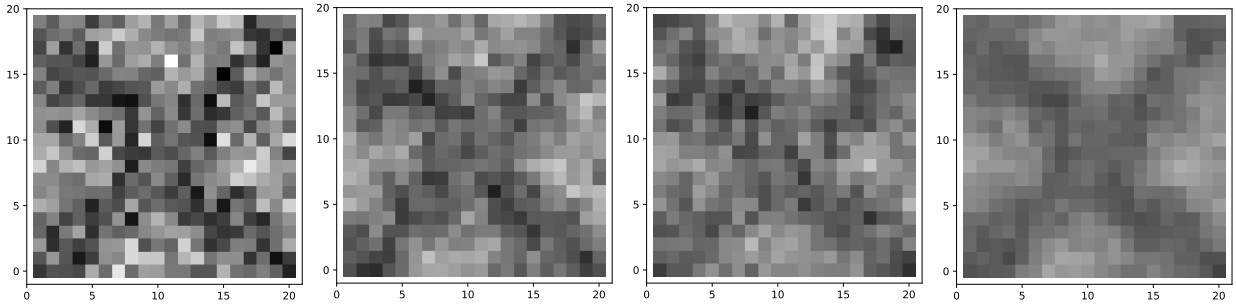
(a) $\mathbf{x}^{(0)}$ (b) $\mathbf{x}^{(1)}$ (c) $\mathbf{x}^{(100)}$ (d) Mean Image

Figure 6: Images obtained from a Gibbs sampler ($\sigma = 2$ and a second-order neighborhood), with $\mathbf{x}^{(0)}$ as the observed data image, $\mathbf{x}^{(1)}$, and $\mathbf{x}^{(100)}$ as result from the first and last iteration.



(a) $\mathbf{x}^{(0)}$ (b) $\mathbf{x}^{(1)}$ (c) $\mathbf{x}^{(100)}$ (d) Mean Image

Figure 7: Images obtained from a Gibbs sampler ($\sigma = 15$ and a second-order neighborhood), with $\mathbf{x}^{(0)}$ as the observed data image, $\mathbf{x}^{(1)}$, and $\mathbf{x}^{(100)}$ as result from the first and last iteration.



(a) Data Image (b) $\mathbf{x}^{(1)}$ (c) $\mathbf{x}^{(100)}$ (d) Mean Image

Figure 8: Images obtained from a Gibbs sampler ($\sigma = 5$ and a first-order neighborhood), with $\mathbf{x}^{(0)} = 57.5$ as the observed data image, $\mathbf{x}^{(1)}$, and $\mathbf{x}^{(100)}$ as result from the first and last iteration. Note, (a) shows the data image that was observed, not the the initial starting image that was used for the Gibbs sampler. The presence of (a) here is just for comparison with images generated by the model in our project.

4 Discussion

From the above result, we are able to restore the original image quite well. In particular, $\sigma = 5$ works the best, and first-order neighborhood is slightly better than second-order. The

initial $\mathbf{x}^{(0)} = 57.5$ works similar as $\mathbf{x}^{(0)}$ as the observed data. The mean image in every case is pretty smooth since it is averaged over 100 images. σ seems to be a more important factor to the sampled images compared with neighborhood structure. In the conditional posterior distribution, variance is equal to $\frac{\sigma^2}{v_i+1}$. When σ changes, the variance changes in the order of σ^2 (when the neighborhood structure is fixed). But when neighborhood structure changes and σ is fixed, the variance only changes by a factor of ($\frac{1}{5} \sim \frac{1}{9}$). Although the mean of the conditional posterior also changes when neighborhood structure changes, but it depends on the data, and the order of the difference between mean is only a linear combination of y_i and $\bar{x}_{\delta_i}$. In this sense, σ is a more significant factor. Our results have also suggested this idea. There is not a significant difference between Gibbs saplers with different initial starting image $\mathbf{x}^{(0)}$, indicating the Markov chain moves fast from the initial $\mathbf{x}^{(0)}$ and converges to roughly the same result after some iterations.

5 Contribution

Since this is a group project, we are working on different parts of it. Here is the detailed work we have done separately:

- Niloufar Dousti Mousavi: Draft of model proof in deviriation and appendix.
- Wangfei Wang: Project report draft, experiment analysis, final review and proofread of the final version.
- Yanzi Jin: Code for the experiment, reorganize the repoer after the first draft, reformulate the proof and correct the typos in the proof in the derivation and appendix.

6 Appendix

In the appendix, we provide detailed derivation of the univariate conditional posterior distribution used for Gibbs sampling for the project. For the purpose of generation, we consider derivation on multivariate Gaussian, which could be easily applied to univariate case. Part of the derivation was obtained in [6].

6.1 Conditional Gaussian Distributions

One important property of Gaussian distribution is that if two sets of variables are jointly Gaussian, then the conditional distribution of one set conditioned on the other is again Gaussian. Suppose $\mathbf{x}$ is a D-dimenstional vector and follows Gaussian distribution $\mathcal{N}(\mu, \Sigma)$, where μ and Σ are the mean and covariance matrix of $\mathbf{x}$.
We partition $\mathbf{x}$ into two disjoint subsets $\mathbf{x}_a$ and $\mathbf{x}_b$, suppose $\mathbf{x}_a$ is the first M component of $\mathbf{x}$ and $\mathbf{x}_b$ is the remaining $D - M$ components. Define $\Lambda = \Sigma^{-1}$ as the precision matrix, which is the inverse of the covariance matrix Σ. Then we have the corresponding mean vector μ and precision matrix

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix} \quad \mu = \begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix} \quad \Sigma = \begin{pmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ab}^T & \Sigma_{bb} \end{pmatrix} \quad \Lambda = \Sigma^{-1} = \begin{pmatrix} \Lambda_{aa} & \Lambda_{ab} \\ \Lambda_{ab}^T & \Lambda_{bb} \end{pmatrix} \tag{6}$$

Note that since Σ is symmetric, Λ_{aa} and Λ_{bb} are also symmetric, when $\Lambda_{ab}^T = \Lambda_{ba}$.
To find the conditional distribution of $f(\mathbf{x}_a|\mathbf{x}_b)$, we could start with the joint distribution $f(\mathbf{x}) = f(\mathbf{x}_a, \mathbf{x}_b)$. Making use of the partition in equation (6), we obtain

$$\begin{aligned} -\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu) = &-\frac{1}{2}(\mathbf{x}_a-\mu_a)^T\Lambda_{aa}(\mathbf{x}_a-\mu_a) - \frac{1}{2}(\mathbf{x}_a-\mu_a)^T\Lambda_{ab}(\mathbf{x}_b-\mu_b) \\ &-\frac{1}{2}(\mathbf{x}_b-\mu_b)^T\Lambda_{ba}(\mathbf{x}_a-\mu_a) - \frac{1}{2}(\mathbf{x}_b-\mu_b)^T\Lambda_{bb}(\mathbf{x}_b-\mu_b) \end{aligned} \tag{7}$$

If we see equation (7) as a function of $\mathbf{x}_a$, it is a quadratic form. Therefore we the conditional probability $f(\mathbf{x}_a|\mathbf{x}_b)$ will be a Gaussian.
Note

$$\mathcal{N}(\mu, \Sigma) \propto \exp\left[-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\right] \tag{8}$$

We only consider the quadratic form in the exponent of the distribution, by expanding it and using the symmetry of Σ, we have

$$-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu) = -\frac{1}{2}\mathbf{x}^T\Sigma^{-1}\mathbf{x} + \mathbf{x}^T\Sigma^{-1}\mu + \text{const} \tag{9}$$

where 'const' are the terms that are independent of $\mathbf{x}$. As a common operation, sometimes called 'completing the square', we can obtain the covariance matrix by looking at the quadratic term $-\frac{1}{2}\mathbf{x}^T\Sigma^{-1}\mathbf{x}$ and the second order term of the right side of equation (9), which is $\mathbf{x}^T\Sigma^{-1}\mu$, we can get the mean μ. Using the same trick for $f(\mathbf{x}_a|\mathbf{x}_b)$ in equation (7), we fix

$\mathbf{x}_a$ and treat $\mathbf{x}_b$ as constant. We can obtaind the covariance matrix $\Sigma_{a|b}$ from the quadratic term

$$-\frac{1}{2}\mathbf{x}_a^T\Lambda_{aa}\mathbf{x}_a \;\Rightarrow\; \Sigma_{a|b} = \Lambda_{aa}^{-1}. \tag{10}$$

Next look at the remaining terms that are linear in $\mathbf{x}_a$ in equation (7),

$$\mathbf{x}_a^T[\Lambda_{aa}\mu_a - \Lambda_{ab}(\mathbf{x}_b - \mu_b)] \tag{11}$$

the coefficient of $\mathbf{x}_a$ must equal $\Sigma_{a|b}^{-1}\mu_{a|b}$:

$$\begin{aligned}\mu_{a|b} &= \Sigma_{a|b}[\Lambda_{aa}\mu_a - \Lambda_{ab}(\mathbf{x}_b - \mu_b)] \\ &= \mu_a - \Lambda_{aa}^{-1}\Lambda_{ab}(\mathbf{x}_b - \mu_b)\end{aligned} \tag{12}$$

So far we have the mean vector and the covariance matrix in the form of the partitioned precision matrix of the original joint distribution $f(\mathbf{x}_a, \mathbf{x}_b)$, we can express them in the form of the original partitioned covariance matrix. Using Schur Complement Theory:

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} M & -MBD^{-1} \\ -D^{-1}CM & D^{-1} + D^{-1}CMBD^{-1} \end{pmatrix} \tag{13}$$

where $M = (A - BD^{-1}C)^{-1}$, which is the *Schur complement* of the submatrix D on the left side of equation (13).
Using the above Schur Complement Theory, we have

$$\begin{aligned}\Lambda_{aa} &= (\Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba})^{-1} \\ \Lambda_{aa} &= -(\Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba})^{-1}\Sigma_{ab}\Sigma_{bb}^{-1}.\end{aligned} \tag{14}$$

Finally we get the mean vector and covariance matrix of $f(\mathbf{x}_a|\mathbf{x}_b)$ in the form of original covariance matrix.

$$\begin{aligned}\mu_{a|b} &= \mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(\mathbf{x}_b - \mu_b) \\ \Sigma_{a|b} &= \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}\end{aligned} \tag{15}$$

6.2 Bayes' Rule for Gaussian Variables

Suppose $\mathbf{x}$ and $\mathbf{y}$ are two D-dimenstional vectors and follow

$$f(\mathbf{x}) = \mathcal{N}\left(\mu, \Lambda^{-1}\right), \quad f(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{x}, L^{-1}), \tag{16}$$

where μ is the mean of $\mathbf{x}$, Λ is the precision matrix, which is the inverse of the covariance matrix Σ. Similarly, L is the precision matrix of $\mathbf{y}$ conditioned on $\mathbf{x}$.
Now assume

$$\mathbf{y} = A\mathbf{x} + \mathbf{b}, \quad \mathbf{x} \sim \mathcal{N}(\mu, \Lambda^{-1}), \quad \mathbf{y}|\mathbf{x} \sim \mathcal{N}(A\mathbf{x} + \mathbf{b}, L^{-1}). \tag{17}$$

We want to find $f(\mathbf{x}|\mathbf{y})$, first find an expression for the joint distribution over $\mathbf{x}$ and $\mathbf{y}$. Now we define $\mathbf{z} = \begin{pmatrix}\mathbf{x} & \mathbf{y}\end{pmatrix}^T$ and consider the log of the joint distribution

$$\begin{aligned}\log f(\mathbf{z}) =& \log f(\mathbf{x}) + \log f(\mathbf{y}|\mathbf{x}) \\ =& -\frac{1}{2}(\mathbf{x}-\mu)^T\Lambda(\mathbf{x}-\mu) \\ & -\frac{1}{2}(\mathbf{y}-A\mathbf{x}-\mathbf{b})^T L(\mathbf{y}-Ax-\mathbf{b}) + \text{const}\end{aligned} \tag{18}$$

where the 'const' term is independent of $\mathbf{x}$ and $\mathbf{y}$. Since this is a quadractic function of the component $\mathbf{z}$, $f(\mathbf{z})$ should be a Gaussian distribution. To find the precision matrix, we consider the second term in the above equation (18).

$$\begin{aligned}& -\frac{1}{2}(\mathbf{y}-A\mathbf{x}-\mathbf{b})^T L(\mathbf{y}-Ax-\mathbf{b}) \\ =& -\frac{1}{2}\mathbf{x}^T(\Lambda + A^T LA)\mathbf{x} - \frac{1}{2}\mathbf{y}^T L\mathbf{y} + \frac{1}{2}\mathbf{x}^T(A^T L)\mathbf{y} + \frac{1}{2}\mathbf{y}^T(LA)\mathbf{x} \\ =& -\frac{1}{2}\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix}^T \underbrace{\begin{pmatrix}\Lambda + A^T LA & -A^T L \\ -LA & L\end{pmatrix}}_{\mathrm{R}} \begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix} \\ =& -\frac{1}{2}\mathbf{z}^T R\mathbf{z}.\end{aligned} \tag{19}$$

So the Gaussian distribution over $\mathbf{z}$ has precision matrix given by

$$R = \begin{pmatrix}\Lambda + A^T LA & -A^T L \\ -LA & L\end{pmatrix}.$$

The covariance is found by taking the inverse of R

$$\mathrm{cov}(\mathbf{z}) = R^{-1} = \begin{pmatrix}\Lambda^{-1} & \Lambda^{-1}A^T \\ A\Lambda^{-1} & L^{-1} + A\Lambda^{-1}A^T\end{pmatrix} \tag{20}$$

Similarly we can find the mean of Gaussian distribution over $\mathbf{z}$ by identifying the linear term in equation (18)

$$\mathbf{x}^T\Lambda\mu - \mathbf{x}^T A^T L\mathbf{b} + \mathbf{y}^T L\mathbf{b} = \begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix}^T \begin{pmatrix}\Lambda\mu - A^T L\mathbf{b} \\ L\mathbf{b}\end{pmatrix} \tag{21}$$

We can obtain the mean of $\mathbf{z}$

$$\mathbb{E}[\mathbf{z}] = R^{-1}\begin{pmatrix}\Lambda\mu - A^T L\mathbf{b} \\ L\mathbf{b}\end{pmatrix}^T = \begin{pmatrix}\mu \\ A\mu + \mathbf{b}\end{pmatrix}. \tag{22}$$

Next we find an expression for the marginal distrbution $f(\mathbf{y})$ by marginalizing $\mathbf{x}$. The mean could be obtained by looking at the linear term, similarly with what we did in equation (9).

$$\mathbb{E}[\mathbf{y}] = A\mu + \mathbf{b} \tag{23}$$

$$\mathrm{cov}[\mathbf{y}] = R^{-1}_{2,2} = L^{-1} + A\Lambda^{-1}A^T. \tag{24}$$

Finally we get expression for $f(\mathbf{x}|\mathbf{y})$ from what we got in equation (12).

$$\begin{aligned} \mathbb{E}[\mathbf{x}|\mathbf{y}] &= (\Lambda + A^T L A)^{-1}(\Lambda\mu + A^T L(\mathbf{y} - \mathbf{b})) \\ \text{cov}[\mathbf{x}|\mathbf{y}] &= (\Lambda + A^T L A)^{-1} \end{aligned} \tag{25}$$

More specifically, in our case, $A = I$, $b = 0$, and $\mathbf{x}$ and $\mathbf{y}$ are univariate variables. We have

$$\begin{aligned} \mathbb{E}[\mathbf{x}|\mathbf{y}] &= (\Lambda + L)^{-1}(\Lambda\mu + L\mathbf{y}) \\ \text{cov}[\mathbf{x}|\mathbf{y}] &= (\Lambda + L)^{-1} \end{aligned} \tag{26}$$

References

[1] Mary Kathryn Cowles and Bradley P. Carlin. Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review. *Journal of the American Statistical Association*, 91(434):883–904, 1996.

[2] Julian Besag and Peter J. Green. Spatial statistics and Bayesian computation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 55(1):25–37, 1993.

[3] Julian Besag, Jeremy York, and Annie Mollie. Bayesian Image Restoration, with Two Applications in Spatial Statistics. *Ann Inst Stat Math*, 43(1):1–59, 1991.

[4] John M Hammersley and Peter Clifford. Markov Fields on Finite Graphs and Lattices, 1971.

[5] Geof H. Givens and Jennifer A. Hoeting. *Computational Statistics, Second Edition*. John Wiley & Sons, Inc., Hoboken, New Jersey.

[6] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.